Remarks

I. Status of claims

Claims 1-9, 11-25, and 27-30 are pending.

II. Claim rejection under 35 U.S.C. § 102

The Examiner has rejected claim 5 under 35 U.S.C § 102(e) over Turek (U.S. 6,460,070).

The relevant part of 35 U.S.C. § 102(e) states that a person shall be entitled to an invention, unless - "the invention was described in -- (1) an application for patent published under section 122(b), by another filed in the United States before the invention by the applicant for patent..." Anticipation under 35 U.S.C. § 102(e) requires that each and every element of the claimed invention be present, either expressly or inherently, in a single prior art reference. EMI Group N. Am., Inc., v. Cypress Semiconductor Corp., 268 F.3d 1342, 1350 (Fed. Cir. 2001). Anticipation must be proved by clear and convincing evidence. Electro Medical Systems, S.A. v. Cooper Life Sciences, Inc., 34 F3d 1048, 1052 (Fed. Cir. 1994).

Independent claim 5 recites:

> 5. A system for managing a plurality of distributed nodes of a network, comprising:
>
> a recovery module configured to migrate from one network node to another, determine a status of a network node, and initiate a recovery process on a network node having one or more failed node processes, wherein the recovery module is configured to determine the status of a network node in accordance with a heartbeat messaging protocol.

In support of the rejection of claim 5, the Examiner has stated that (emphasis added):

> As per claim 5 Turek disclosed a system for managing a plurality of distributed nodes of a network, comprising: a recovery modules configured to migrate from one network node to another, determine a status of a network, and initiate a recovery process on a failed network node (col.2, lines 65-67 & col.2, lines 1-46) wherein the recovery module is configured to

> determine the status of a network node in accordance with a heartbeat messaging protocol (col.2, lines 22-46). Although Turek did not specifically mentioned a heartbeat messaging protocol to determine the status of a network node. However Turek did disclose collecting information about network conditions to include network node by the use of mobile software agents that that periodically check the network status information, which is an inherent function of a heartbeat messaging protocol.

With regard to "collecting information about network conditions," Turek expressly teaches that "Yet another object of the present invention is to collect information about network conditions as mobile software agents are dispatched and migrated throughout a large computer network to correct network faults, wherein such information is then useful in diagnosing new faults" (col. 2, lines 21-46). Collecting from migratory software agents information useful for diagnosing new faults, however, does not constitute a teaching that each software agent is "configured to determine the status of a network node in accordance with a heartbeat messaging protocol."

Moreover, it is well settled that:

> To serve as an anticipation when the reference is silent about the asserted inherent characteristic, such gap in the reference may be filled with recourse to extrinsic evidence. Such evidence must make clear that the missing descriptive matter is necessarily present in the thing described in the reference, and that it would be so recognized by persons of ordinary skill.

Continental Can Company USA, Inc. v. Monsanto Co., 948 F.2d 1264, 20 USPQ2d 1746 (Fed. Cir. 1991). In the present case, there is no basis for the Examiner to assume that one skilled in the art would have recognized that the software agents deployed by the dispatch mechanism in accordance with Turek's teachings necessarily determines the status of a network node in accordance with a heartbeat messaging protocol.

Turek explicitly teaches that when a particular software agent is received at a given node, the software agent "determines whether the event originated from the node" and, if so, "identifies the cause and, if possible, undertakes a corrective or other action depending on the nature of the event in question" (col. 2, lines 49-53; emphasis added). Based on this disclosure, one skilled in the art at the time the invention was made reasonably would have concluded that the software agents are configured to identify the node on which to attempt a corrective action using some sort of identifier. There is no part of Turek's disclosure that teaches that the software agents use a heartbeat messaging protocol to determine the status of a network node. Indeed, Turek's software agents are deployed only after an event, such as a network fault, has occurred (see, e.g., col. 2, lines 35-37). Therefore, there is no need for the software agents to use a heartbeat messaging protocol to determine whether an event originated from a particular node. Instead, each software agent need only be tailored to specifically identify the particular network fault that triggered the deployment of the software agent by the dispatch mechanism 15 (see, e.g., col. 5, lines 31-60 and col. 6, lines 23-59).

Heartbeat messaging is a well-known feature of clustered networks in accordance with which a "heartbeat" message is sent regularly from one node to another node merely to detect failed applications or failed nodes (see, e.g., Forbes, U.S. 6,728,896, col. 4, lines 27-33, and col. 9, lines 2-14; cited by the Examiner). Such heartbeat messages indicate the state "I'm here, are you here?" (see, e.g., Forbes, col. 9, lines 2-14). There is no reasonable basis for the Examiner's assumption that the software agent that is selected and deployed specifically in response to a particular, previously identified fault event would have used a heartbeat messaging protocol to determine whether that fault event originated from a given node or to identify the cause that fault event.

For at least these reasons, the Examiner's rejection of independent claim 5 under 35 U.S.C. § 102(e) over Turek should be withdrawn.

III. Claim rejections under 35 U.S.C. § 103

A. Independent claim 5

The Examiner has rejected claim 5 under 35 U.S.C. § 103(a) over Turek in view of Harvell (U.S. 6,834,302). In particular, the Examiner has stated that:

> As per claims 5 Turek disclosed a system for managing a plurality of distributed nodes of a network, comprising: a recovery modules configured to migrate from one network node to another, determine a status of a network, and initiate a recovery process on a network node having one or more failed node processes (col. 2, lines 65-67 & col. 2, lines 1-46) wherein the recovery module is configured to determine the status of a network node in accordance with a heartbeat messaging protocol (col.2, lines 22-46). However Turek did not specifically mentioned a heartbeat messaging protocol to

> determine the status of a network node. In the same field of endeavor Harvell disclosed a heartbeat messaging protocol to determine the status of a network node (col.2, lines 50-56).

The cited section of Harvell's disclosure reads as follows:

> The second protocol requires a client computer in the network associated with the DTG to periodically send a heartbeat message to the primary master name server for the DTG's domain name zone. The heartbeat message is used by the primary master name server to determine if the network associated with the DTG is still connected.

In accordance with this disclosure, a client computer in the network associated with a DTG (dynamic topology group) periodically sends a heartbeat message that is used to determine if the network associated with the DTG is still connected. Contrary to the Examiner's statement, this disclosure has nothing whatsoever to do with determining the status of a node; instead, it relates to the one-way transmission of heartbeat messages from a client node to determine whether a network associated with that node is still connected. Consequently, there is nothing in this disclosure that would have led one skilled in the art to modify Turek's migratory agents to determine the status of a network node in accordance with a heartbeat messaging protocol.

In addition, in support of the proposed combination of Turek and Harvell, the Examiner has stated that:

> It would have been obvious to one in the ordinary skill in the art at the time the invention was made to have incorporated the heartbeat messaging protocol to determine the status of a network node as disclosed by Harvell in the a system for managing a plurality of distributed nodes of a network as disclosed by Turek in order to make the managing system more reliable and responsive resulting in determining accurate diagnosis and status of the network nodes.

Neither Turek nor Harvell discloses anything that would have led one skilled in the art at the time the invention was made to the rationale (i.e., "to make the managing system more reliable and responsive resulting in determining accurate diagnosis and status of the network nodes") on which the Examiner's rejection of claim 5 is premised. Thus, the Examiner does not have any reasonable basis for concluding that such a person would have been motivated whatsoever to combine the references in the proposed manner.

For at least these reasons, the rejected of claim 5 under 35 U.S.C. § 103(a) over Turek and Harvell should be withdrawn.

B. Claims 1-4, 6-9, and 11-25

The Examiner has rejected claims 1-4, 6-9, and 21-25 under 35 U.S.C § 102(e) over Turek (U.S. 6,460,070) in view of Sreenivasan (U.S. 2002/0049845).

1. Independent claim 1

Independent claim 1 recites:

> 1. A system for managing a plurality of distributed nodes of a network, comprising:
>
> a network management module that launches migratory recovery modules into the network to monitor status of each of the network nodes;
>
> wherein each of the recovery modules is configured to migrate from one network node to another, determine a respective status of each of the network nodes to which it has migrated, and initiate a recovery process on ones of the network nodes having one or more failed node processes, the recovery modules determine the status of each of the network nodes, and the network management module monitors transmissions that are received from the recovery modules to provide periodic monitoring of the status of each of the network nodes.

In support of the rejection of independent claim 1, the Examiner has stated that:

> As per claims 1, 11, 19 & 20 Turek-Sreenivasan disclosed a method for managing a plurality of distributed nodes of a network, comprising: a network management module that launches migratory recovery modules into the network to monitor status of each of the network nodes; wherein each of the recovery modules is configured to migrate from one network to another, determine a respective status of each of the network nodes to which it has migrated, and initiate a recovery process on failed ones of the network nodes.(col.3, lines 48-64,

> col.1, lines 59-62, 65-67, col.2, lines 22-26, col.2, lines 1-3, col.2, lines 22-26 & col.5, lines 32-60), having one or more failed node processes, the recovery modules determine the status of each of the network nodes, and the network management module monitors transmissions that are received from the recovery modules to provide periodic monitoring of the status of the network nodes (col.7, lines 58-67 & col.8, lines 1-9). However Turek did not explicitly disclose the recovery module (software agents) periodically sending network node status. In the same field of endeavor Sreenivasan disclosed recovery modules sending periodic status updates of a specific node to the other network entity node (page.2, paragraph.26 & page.6, paragraphs. 111 & 112).

a. The Examiner's characterization of Turek's disclosure is incorrect

Contrary to the Examiner's statement, Turek's system does not launch migratory recovery modules into a network to monitor status of each of the network nodes, wherein the recovery modules determine the status of each of the network nodes and the network management module monitors transmissions that are received from the recovery modules to provide periodic monitoring of each of the network nodes.

i. Turek's disclosure

In accordance with Turek's disclosure, the mobile software agents are deployed by the dispatch mechanism 15 only in response to either a report of a "network 'fault', alarm or other such trigger" (col. 7, lines 3-4) or a "request for maintenance in some non-specified area of the network" (col. 7, line 7). The dispatch mechanism 15 deploys a selected one of the software agents to the particular location of a fault or a particular area of a network where the fault is likely to have occurred (see, e.g., col. 7, lines 1-57). If the initial given node location does not contain the specific fault for which the software agent was deployed, the software agent identifies "a subset of nodes (associated with the given node) that remain candidates for locating the error" (col. 8, lines 31-32). Once the particular fault is located and diagnosed, the software agent attempts to fix the problem (see col. 9, lines 21-22). "If unable to effect repairs, the agent will, at a minimum, report back with the diagnosis to a user interface of the dispatch mechanism" (col. 9, lines 28-30). Turek does not teach or suggest anything that that would have led one skilled in the art at the time the invention was made to believe that the software agent migrates to any other nodes after attempting to "effect repairs" and reporting back with the diagnosis. Indeed, in accordance with Turek's teachings, each of the mobile software agents is deployed to diagnose and, if possible correct, only one particular network fault. Therefore, there is no need whatsoever for any of Turek's software agents to migrate from the node that contains the particular network fault that the software agent was deployed to diagnose and correct.

Thus, one skilled in the art at the time the invention was made would not have had any reasonable basis for believing that Turek's system launches migratory recovery modules into a network to monitor status of each of the network nodes, as recited in claim 1. In accordance with its ordinary and accustomed meaning, the verb "monitor" means "to watch, keep track of, or check usu. for a special purpose" (Merriam-Webster's Collegiate Dictionary, Tenth Edition (1995). The dispatch mechanism 15 does not launch migratory recovery modules into the network to monitor status of each of the network nodes. Instead, the dispatch mechanism 15 deploys the software agents only after a network fault has been reported (see, e.g., col. 7, lines 3-4) or after receiving a "request for maintenance in some non-specified area of the network" (col. 7, line 7), and the dispatched agents cease migrating after reaching their respective target nodes.

In addition, one skilled in the art at the time the invention was made would not have had any reasonable basis for believing that Turek's software agents determine the status of each of the nodes of a network. To the contrary, based on Turek's teaching, such a person reasonably would have recognized that the software agents are designed to recursively narrow the search for the particular network nodes for which they were respectively deployed and, after arriving at the target network nodes, the software agents attempt to effect repairs and report back diagnoses; the dispatched agents cease migrating after reaching their respective target nodes. That is, the software agents are not configured to determine the status of each of the network nodes, as recited in claim 1. Moreover, Turek only teaches that each of the software agents is configured to determine whether a particular event originated from a node (see col. 2, lines 49-53); Turek does not teach that these agents are configured to determine the status of their respective target nodes. Consequently, the dispatch mechanism 15 is not configured to provide periodic monitoring of the status of each of the network nodes, as recited in claim 1.

ii. The cited sections of Turek's disclosure do not support the Examiner's position

The Examiner has pointed to col.3, lines 48-64, col.1, lines 59-62, 65-67, col.2, lines 22-26, col.2, lines 1-3, col.2, lines 22-26 & col.5, lines 32-60 in support of the position that Turek discloses "a network management module that launches migratory recovery modules into the network to monitor status of each of the network nodes; wherein each of the recovery modules is configured to migrate from one network node to another, determine a respective status of each of the network nodes to which it has migrated, and initiate a recovery process on ones of the network nodes having one or more failed node processes, the recovery modules determine the status of each of the network nodes" (see ¶ 12 of the Office action). The cited sections of Turek's disclosure, however, do not support the Examiner's characterization of Turek's teachings.

Col.3, lines 48-64:

Col.3, lines 48-64 recites:

> Referring now to FIG. 1, the invention is preferably implemented in a large distributed computer environment 10 comprising up to thousands of "nodes." The nodes will typically be geographically dispersed and the overall environment is "managed" in a distributed manner. Preferably, the managed environment (ME) is logically broken down into a series of loosely-connected managed regions (MR) 12, each with its own management server 14 for managing local resources with the MR. The network typically will include other servers (not shown) for carrying out other distributed network functions. These include name servers, security servers, file servers, threads servers, time servers and the like. Multiple servers 14 coordinate activities across the enterprise and permit remote site management and operation. Each server 14 serves a number of gateway machines 16, each of which in turn support a plurality of endpoints 18. The server 14 coordinates all activity within the MR using a terminal node manager 20.

This disclosure does not teach or suggest "a network management module that launches migratory recovery modules into the network to monitor status of each of the network nodes; wherein each of the recovery modules is configured to migrate from one network node to another, determine a respective status of each of the network nodes to which it has migrated, and initiate a recovery process on ones of the network nodes having one or more failed node processes, the recovery modules determine the status of each of the network nodes."

Col. 1, lines 59-62, 65-67:

Col.1, lines 59-62, 65-67 recites:

> It would be a significant advantage to provide some automatic means of diagnosing and correcting network problems in this type of computer environment. The present invention addresses this important problem.
>
> . . .
>
> It is a primary object of this invention to automatically diagnose faults or other events that occur in a large, distributed computer network.

This disclosure does not teach or suggest "a network management module that launches migratory recovery modules into the network to monitor status of each of the network nodes; wherein each of the recovery modules is configured to migrate from one network node to another, determine a respective status of each of the network nodes to which it has migrated, and initiate a recovery process on ones of the network nodes having one or more failed node processes, the recovery modules determine the status of each of the network nodes."

It is noted that diagnosing and correcting network problems does not constitute monitoring the status of each of nodes in a network.

Col.2, lines 1-3:

Col.2, lines 1-3 recites that "It is another primary object of this invention to deploy a software "agent" into a distributed computer network environment to diagnose and, if possible, correct a fault."

This disclosure does not teach or suggest "a network management module that launches migratory recovery modules into the network to monitor status of each of the network nodes; wherein each of the recovery modules is configured to migrate from one network node to another, determine a respective status of each of the network nodes to which it has migrated, and initiate a recovery process on ones of the network nodes having one or more failed node processes, the recovery modules determine the status of each of the network nodes."

It is noted that diagnosing and correcting network problems does not constitute monitoring the status of each of nodes in a network.

Col.2, lines 22-26:

Col.2, lines 22-26 recites:

> Yet another object of the present invention is to collect information about network conditions as mobile software agents are dispatched and migrated throughout a large computer network to correct network faults, wherein such information is then useful in diagnosing new faults.

This disclosure does not teach or suggest "a network management module that launches migratory recovery modules into the network to monitor status of each of the network nodes; wherein each of the recovery modules is configured to migrate from one network node to another, determine a respective status of each of the network nodes to which it has migrated, and initiate a recovery process on ones of the network nodes having one or more failed node processes, the recovery modules determine the status of each of the network nodes."

It is noted that collecting from mobile software agents information that is useful in diagnosing new faults does not constitute monitoring the status of each of nodes in a network.

Col.5, lines 32-60

Col.5, lines 32-60 recites:

> A preferred embodiment of the present invention is implemented in the enterprise environment illustrated above. In this embodiment, a set of "software agents" are available at a central location (e.g., manager 14) or at a plurality of locations (e.g., the gateways 16) in the o network where network errors are reported. The software agents are "mobile" in the sense that the agents are dispatched (as will be described below) from a dispatch mechanism and then migrate throughout the network environment. Generally, the mobile software agents traverse the network to diagnose and, if possible, to correct a network fault.

> Thus, when a network error or "fault" is reported whose cause and location are not apparent or readily ascertainable, an appropriate agent is identified and dispatched to determine this information. Preferably, the agent is dispatched to the actual node in the network at which the fault condition occurs. As will be seen, the particular error, as well as other associated events, generally provide a "clue" or clues regarding the network location to which the agent should be sent, as well as the type of agent to send. If the agent does not find the fault at the initial location to be examined, the agent then migrates through the network to locate the error. The agent preferably chooses its path through the network based on the information received at the dispatching location, as well as information gleaned from each examined location. As will be seen, the particular "path" typically varies as the software agent migrates through the network because information gleaned from a particular node may redirect the agent in some given manner.

This disclosure does not teach or suggest "a network management module that launches migratory recovery modules into the network to monitor status of each of the network nodes; wherein each of the recovery modules is configured to migrate from one network node to another, determine a respective status of each of the network nodes to which it has migrated, and initiate a recovery process on ones of the network nodes having one or more failed node processes, the recovery modules determine the status of each of the network nodes."

It is noted that diagnosing and correcting network problems does not constitute monitoring the status of each of nodes in a network.

b. Sreenivasan's disclosure does not make-up for the failure of Turek to teach or suggest "the network management module monitors transmissions that are received from the recovery modules to provide periodic monitoring of the status of each of the network nodes," as proposed by the Examiner

The Examiner has stated that:

> "... Turek did not explicitly disclose the recovery module (software agents) periodically sending network node status. In the same field of endeavor Sreenivasan disclosed recovery modules sending periodic status updates of a specific node to the other network entity node (page 2, paragraph 26 & page 6, paragraphs 111 & 112).

Contrary to the Examiner's position, however, Sreenivasan does not disclose anything about recovery modules of the type disclosed in Turek, much less anything about such modules "sending periodic status updates of a specific node to the other network entity node."

Paragraph 26 reads as follows (emphasis added):

> [0026] To address the problems stated above, and to solve other problems which will become apparent in reading the specification and claims, a high availability computing system and method are described. The high availability computing system includes a plurality of computer nodes (for example, a server system) connected by a first and a second network, wherein the computer nodes communicate with each other to detect server failure and transfer applications to other computer nodes on detecting server failure.

Thus, in accordance with the teachings of ¶ 26, recovery modules of the type disclosed in Turek are not used to periodically send network node status; instead, "the computer nodes communicate with each other to detect server failure."

Paragraphs 111 and 112 read as follows:

> [0111] As noted above, the main component of the Cluster Membership Service is the Cluster Membership Daemon. In some embodiments, the Cluster Membership Daemon is responsible for running the whole protocol and is represented by the Membership Daemon that runs on it. The daemon maintains in an internal variable its current view of the Membership. The daemon is said to have delivered a new Membership when the value of that variable is changed.
>
> [0112] Each CMD sends messages to other CMD's by invoking a broadcast primitive. The destination of the broadcast are all the nodes in S except the originator. Typically, the broadcast primitive is the only way CMD sends messages. The semantic of the broadcast primitive are very weak. Message can be lost and there are little guarantees on the ordering at the receive end. Current implementation of the daemon uses UDP/IP, however any datagram transport can be substituted. The broadcast primitive prepends a header to the message. As stated above CMD uses one type of message. Each message contains useful information and at the same time can be considered as an "I'm alive message" from the sender. CMD is required to periodically broadcast a message. The interval between broadcasts is a configurable parameter.

Thus, neither ¶ 111 nor ¶ 112 teaches that recovery modules of the type disclosed in Turek are not used to periodically send network node status. Instead, the CMD processes running on each of the servers in the cluster communicate with each other to detect server failure. In particular, Sreenivasan teaches that each server 12 in the cluster runs Cluster Management Services (CMS) 32 and Group Communication Services (GCS) 34 (see ¶ 79), where an instance of the CMS service 12 is referred to as a Cluster Management Daemon (CMD) (see ¶ 85). Nodes are represented by the CMD processes that run on them and the failure of such a CMD is interpreted as the failure of the node (see ¶ 85). The CMD processes running on the servers 12 communicate using a Cluster Management Protocol 36 that includes an initialization phase, a monitoring phase, and an agreement phase (see ¶¶ 85-88). During the monitoring phase, the nodes in the cluster send and receive heartbeat messages (see ¶ 89). Paragraphs 111 and 112 explain some of the details of the communications between the CMS processes running on the network nodes.

To summarize, the Examiner has taken the position that the CMD processes running on the servers of the Sreenivasan's cluster constitute "recovery modules." These CMD processes, however, are not mobile.

In an effort to make-up for the failure of Turek to teach or suggest "the network management module monitors transmissions that are received from the recovery modules to provide periodic monitoring of the status of each of the network nodes," the Examiner has reasoned that:

> It would have been obvious to one in the ordinary skill in the art at the time the invention was made to have incorporated a recovery module with periodic status update capability as disclosed by Sreenivasan in the system of managing a plurality of distributed nodes of a network in order to make the managing system more reliable and responsive resulting in determining accurate diagnosis and status of the network nodes.

This reasoning, however, does not establish that claim 1 that one skilled in the art would have been led by the teachings of Turek and Sreenivasan to a network management module that "monitors transmissions that are received from the recovery modules to provide periodic monitoring of the status of each of the network nodes," as recited in claim 1.

First, there is nothing in the disclosure of either Turek or Sreenivasan that would have led one skilled in the art at the time the invention was made to modify Turek's migratory software agents to run the server-node-based CMD processes (i.e., the instances of the Cluster Management Services (CMS) 32; see ¶ 85) disclosed in Sreenivasan.

Second, such a person would not have any reasonable basis for believing that Turek's migratory agents possibly could be configured to run the CMD processes.

Third, even assuming for the purposes of argument that Turek's migratory agents could be modified to run the CMD processes, one skilled in the art would not have had a reasonable basis for believing that the CMD framework would work when running on migratory software agents of the type described in Turek.

Fourth, neither Turek nor Sreenivasan teaches or suggests that the modifying Turek's software agents to run the CMD processes disclosed in Sreenivasan would "make the managing system more reliable and responsive resulting in determining accurate diagnosis and status of the network nodes," as asserted by the Examiner. There simply is no reasonable basis whatsoever for the Examiner's assertion that one skilled in the art at the time the invention was made would have recognized that the Examiner's proposed modification of Turek based on Sreenivasan's teachings would "make the managing system more reliable and responsive resulting in determining accurate diagnosis and status of the network nodes."

Thus, contrary to the Examiner's position, one skilled in the art at the time the invention was made would not have been led to modify Turek's migratory software agents to run the CMD processes (i.e., the instances of the Cluster Management Services (CMS) 32; see ¶ 85) disclosed in Sreenivasan.

c. Conclusion

As explained above, Turek's system does not launch migratory recovery modules into a network to monitor status of each of the network nodes, wherein the recovery modules determine the status of each of the network nodes and the network management module monitors transmissions that are received from the recovery modules to provide periodic monitoring of each of the network nodes, as recited in claim 1. Sreenivasan does not make-up for this failure. For at least this reason, the Examiner's rejection of claim 1 under 35 U.S.C. § 103(a) over Turek in view of Sreenivasan should be withdrawn.

Sreenivasan also does not make-up for the failure of Turek to teach or suggest "the network management module monitors transmissions that are received from the recovery modules to provide periodic monitoring of the status of each of the network nodes," as recited in claim 1. For at least this additional reason, the Examiner's rejection of independent claim 1 under 35 U.S.C. § 103(a) over Turek and Sreenivasan should be withdrawn.

2. Claims 2-4, 6-9, 21-25, and 30

Each of claims 2-4, 6-9, 21-25, and 30 incorporates the features of independent claim 1 and therefore is patentable over Turek for at least the same reasons explained above.

3. Independent claim 11

Independent claim 11 recites:

> 11. A method for managing a plurality of distributed nodes of a network, comprising:
>
> (a) on a current one of the network nodes, determining a status of the current network node;
>
> (b) in response to a determination that the current network has one or more failed node processes, initiating a recovery process on the current network node;
>
> (c) after initiating the recovery process, migrating from the current network node to a successive one of the network nodes; and
>
> (d) repeating (a), (b), and (c) with the current network node corresponding to the successive network node for each of the nodes in the network.

In support of the rejection of independent claim 1, the Examiner has stated that:

> As per claims 1, 11, 19 & 20 Turek-Sreenivasan disclosed a method for managing a plurality of distributed nodes of a network, comprising: a network management module that launches migratory recovery modules into the network to monitor status of each of the network nodes; wherein each of the recovery modules is configured to migrate from one network to another, determine a respective status of each of the network nodes to which it has migrated, and initiate a recovery process on failed ones of the network nodes.(col.3, lines 48-64, col.1, lines 59-62, 65-67, col.2, lines 22-26, col.2, lines 1-3, col.2, lines 22-26 & col.5, lines 32-60), having one or more failed node processes, the recovery modules determine the status of each of the network nodes, and the network

> management module monitors transmissions that are received from the recovery modules to provide periodic monitoring of the status of the network nodes (col.7, lines 58-67 & col.8, lines 1-9). However Turek did not explicitly disclose the recovery module (software agents) periodically sending network node status. In the same field of endeavor Sreenivasan disclosed recovery modules sending periodic status updates of a specific node to the other network entity node (page.2, paragraph.26 & page.6, paragraphs. 111 & 112).

On its face, the Examiner has not established a *prima facie* case that claim 11 is obvious over Turek and Harvell. In particular, the Examiner has not shown that any permissible combination of Turek and Harvell would have led one skilled in the art to modify Turek's software agents to perform elements (c) and (d) of claim 11.

In accordance with Turek's teachings, the mobile software agents do not migrate from a current network node to a successive one of the network nodes after initiating a recovery process on the current network node. Instead, after initiating a recovery process, Turek's mobile software agents merely report the problem and the corrective action that was taken to the dispatch mechanism 15 (see col. 8, lines 6-9; FIG. 4). Turek does not teach or suggest anything that that would have led one skilled in the art at the time the invention was made to believe that the software agent migrates to any other nodes after attempting to "effect repairs" and reporting back with the diagnosis. Indeed, in accordance with Turek's teachings, each of the mobile software agents is deployed to diagnose and, if possible correct, only one particular network fault. Therefore, there is no need whatsoever for any of Turek's software agents to migrate from the node that contains the particular network fault that the software agent was deployed to diagnose and correct.

Harvell does not teach or suggest anything whatsoever about migratory software agents.

Thus, Turek and Harvell, either taken alone or in any permissible combination, do not teach or suggest either element (c) or element (d) of claim 11. For at least these reasons, the Examiner's rejection of independent claim 11 under 35 U.S.C. § 103(a) over Turek and Harvell should be withdrawn.

4. Claims 12-19

Each of claims 12-19 incorporates the features of independent claim 11 and therefore is patentable over Turek for at least the same reasons explained above.

5. Independent claim 20

Claim 20 that the computer program comprises computer-readable instructions for causing a computer to perform operations comprising:

> migrating the computer program from one network node to a series of successive network nodes;
>
> determining a status of a current one of the network nodes to which the computer program has migrated;
>
> in response to a determination that the current network has one or more failed node processes, initiating a recovery process on the current network node; and
>
> after initiating the recovery process on the current network node, migrating from the current network node to a successive one of the network nodes.

Claim 20 is patentable over Turk and Harvell for at least the same reasons explained above in connection with independent claim 11. Accordingly, the Examiner's rejection of independent claim 20 under 35 U.S.C. § 103(a) over Turek and Harvell should be withdrawn.

C. Claims 27-29

The Examiner has rejected claims 27-29 under 35 U.S.C. § 103(a) over Turek in view of Douik (U.S. 6,012,152).

Each of claims 27-29 incorporates the features of independent claim 1. Douik does not make-up for the failure of Turek to teach or suggest the features of independent claim 1 discussed above. Therefore, claims 27-29 are patentable over Turek and Douik for at least the same reasons explained above in connection with independent claim 1.

Claims 27-29 also are patentable over Turek in view of Douik for at the following additional reasons.

1. Claim 27

In support of the rejection of claim 27, the Examiner has stated that (emphasis added):

> ... Turek did not explicitly disclose, wherein the network management module statistically identifies target ones of the network nodes to achieve a specified confidence level of network monitoring reliability, and proactively launches the recovery modules into the network by transmitting respective ones of the recovery modules to the identified target network nodes. In the same field of endeavor Douik disclosed wherein the network management module statistically identifies target ones of the network nodes to achieve a specified confidence level of network monitoring reliability, and launches the recovery modules into the network by transmitting respective ones of the recovery modules to the identified target network nodes (col. 11, lines 64-67 & col. 12, lines 1-19).

As pointed out in the Amendment dated November 27, 2006, claim 27 does not recite that the network management module "identifies target ones of the network nodes to achieve a specified confidence level of network monitoring reliability, and launches the recovery modules into the network by transmitting respective ones of the recovery modules to the identified target network nodes." Instead, claim 27 recites that "the network management module determines a number of the recovery modules needed to achieve a specified network monitoring service level, and launches the determined number of recovery modules into the network to achieve the specified network monitoring service level." Thus, on its face, the Examiner's rejection of claim 27 does not establish a *prima facie* case of obviousness (see MPEP § 706.02(j)).

Moreover, Douik does not teach or suggest anything about migratory recovery modules, much less anything about determining a number of the recovery modules needed to achieve a specified network monitoring service level and launching the determined number of recovery modules into the network to achieve the specified network monitoring service level.

For at least these additional reasons, the Examiner's rejection of claim 27 under 35 U.S.C. § 103(a) over Turek in view of Douik should be withdrawn.

2. Claim 28

In support of the rejection of claim 28, the Examiner has stated that (emphasis added):

> ... Turek did not explicitly disclose, wherein the network management module statistically identifies target ones of the network nodes to achieve a specified confidence level of network monitoring reliability, and proactively launches the recovery modules into the network by transmitting respective ones of the recovery modules to the identified target network nodes. In the same field of endeavor Douik disclosed wherein the network management module statistically identifies target ones of the network nodes to achieve a specified confidence level of network monitoring reliability, and launches the recovery modules into the network by transmitting respective ones of the recovery modules to the identified target network nodes (col. 11, lines 64-67 & col. 12, lines 1-19).

The disclosure on which the Examiner's rejection of claim 28 is premised reads as follows (i.e., col. 11, line 64 - col. 12, line 19):

> In yet another aspect, the present invention is a method of proactively managing software faults in a mobile telecommunications network. The method begins by storing knowledge in a knowledge base, the knowledge including a functional model of the network, fault models, and fault scenarios; monitoring the network for observed events and symptoms; and determining a suspected fault to explain the observed events and symptoms, the determining step comprising comparing the observed events and symptoms with stored performance data and statistics, and analyzing the comparison with the stored knowledge. This is followed by determining whether the suspected fault is a known fault; implementing a preventive solution upon determining that the suspected fault is a known fault; and performing a fault trend analysis upon determining that the suspected fault is not a known fault. This is followed by performing diagnostic tests; determining whether a successful diagnosis was obtained; performing a fault localization process upon determining that a successful diagnosis was obtained, the fault localization process including analyzing relationships between components involved in the diagnosis of the fault; and providing diagnosis and localization information to trouble shooters.

In this disclosure, Douik merely compares observed events to stored performance data and statistics in order to determine a suspected fault to explain the observed events and symptoms. Douik does not even hint that target nodes are identified statistically to achieve a specified confidence level of network monitoring reliability. Moreover, Douik does not teach or suggest anything about migratory recovery modules and launching the recovery modules into the network by transmitting respective ones of the recovery modules to the identified target network nodes.

For at least this additional reason, the Examiner's rejection of claim 28 under 35 U.S.C. § 103(a) over Turek in view of Douik should be withdrawn.

3. Claim 29

Claim 29 recites features that essentially track the pertinent features of claim 28 discussed above. Therefore, claim 29 is patentable over Turek and Doiuk for at least the same reasons explained above in connection with claim 28.

IV. Conclusion

For the reasons explained above, all of the pending claims are now in condition for allowance and should be allowed.

Charge any excess fees or apply any credits to Deposit Account No. 08-2025.

Respectfully submitted,

Date: May 7, 2007

Edouard Garcia
Reg. No. 38,461
Telephone No.: (650) 289-0904

Please direct all correspondence to:

Hewlett-Packard Company
Intellectual Property Administration
Legal Department, M/S 35
P.O. Box 272400
Fort Collins, CO 80528-9599